**Theory Questions**

a) Describe briefly the fundamental characteristics of Structural design patterns.

Structural design patterns aim to simplify the structure of code by identifying the relationships between each other. They use inheritance and interfaces in order to work together as a whole.

b) What is the difference between an Abstract and Concrete class?

A concrete class can be instantiated only if it inherits or provides implementation of all its methods where an abstract class cannot be instantiated because of at least a single method not having been implemented.

A concrete class also cannot have any abstract methods unlike the abstract class.

c) Distinguish clearly between class inheritance and interface inheritance.

Class inheritance is when a class extends a predefined object class in Java, to which when functionality is added creates polymorphism.

An interface inheritance creates a base template for an object class and can also support multiple inherits unlike class inheritance.

d) What is the intent of the Adapter design pattern?

The intent of the adapter design pattern is to allow two normally incompatible interfaces to work together as well as other incompatible classes to work together by converting the interface of a class into one expected by a client.

e) What is the intent of the Decorator design pattern?

The intent of decorator design pattern is to attach new behaviours to objects by placing the objects inside special wrapper objects which contain the behaviours.